МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«МИРЭА – Российский технологический университет» РТУ МИРЭА

РТУ МИРЭА

Институт кибербезопасности и цифровых технологий

Кафедра КБ-2 «Прикладные информационные технологии»

Практическая работа № 6

по дисциплине «Безопасность Операционных систем»

«Netfilter, WAF»

Москва

2022

Цель работы

Изучение межсетевых экранов. Приобретение навыков работы с Iptables и WAF.

Время выполнения работы: 4 академических часа.

Порядок выполнения работы

1. **Установить виртуальные машины.**

Развернуть образ виртуальной машины с Kali Linux.

Скачать образ https://github.com/Virtual-Machines/Debian-VirtualBox либо установить самостоятельно. Из образа debian.ova импортировать виртуальную машину, настроить сетевое взаимодействие.

2. **Iptables, Web application firewall.**

Межсетевой экран

Скорее всего, ранее вы уже сталкивались с таким понятием как межсетевой экран. В ядро Linux встроен свой межсетевой экран, называемый Netfilter. Управление им осуществляется с помощью утилиты Iptables.

Межсетевой экран, сетевой экран, файервол, брандмауэр — комплекс аппаратных или программных средств, осуществляющий контроль и фильтрацию проходящих через него сетевых пакетов в соответствии с заданными правилами. Основной задачей сетевого экрана является защита компьютерных сетей или отдельных узлов от несанкционированного доступа. Также сетевые экраны часто называют фильтрами, так как их основная задача — не пропускать (фильтровать) пакеты, не подходящие под критерии, определённые в конфигурации.

Рассмотрим принцип работы Netfilter. Когда сетевые пакеты попадают в сетевой интерфейс, они после ряда проверок ядром проходят последовательность так называемых цепочек. Пакет обязательно проходит через цепочку PREROUTING, после чего определяется, кому он, собственно, был адресован. Если пакет не адресован локальной системе (в нашем случае серверу), он попадает в цепочка FORWARD, а иначе — в цепочку INPUT, после прохождения которой отдается локальным демонам или процессам. После этого при необходимости формируется ответ, который направляется в цепочку OUTPUT. После цепочек OUTPUT или FORWARD пакет в очередной раз встречается с правилами маршрутизации и направляется в цепочку POSTROUTING. В результате прохождения пакетом цепочек фильтрации несколько раз, проверка его принадлежности определенным критериям осуществляется несколько раз. В соответствии с этими проверками к пакету применяется определенное действие:

- ACCEPT — пакет «принимается» и передается в следующую цепочку.

- DROP — удовлетворяющий условию пакет отбрасывается и не передается в другие таблицы или цепочки.
- REJECT — пакет отбрасывается, но при этом отправителю отправляется ICMP-сообщение, сообщающее об отказе.
- RETURN — пакет возвращается в предыдущую цепочку и продолжает её прохождение начиная со следующего правила

Основные команды Iptables

Параметр	Описание	Пример
–append (-A)	Позволяет добавить в указанную цепочку и таблицу заданное правило, помещаемое в КОНЕЦ списка	iptables -A FORWARD критерии -j действие
–delete (-D)	Позволяет удалить заданное номером или каким-либо правилом правило. В первом примере удаляются все правила с номерами 10,12 во всех цепочках, в таблицах filter.	iptables -D 10,12 iptables -t mangle -D PREROUTING критерии -j действие
–rename-chain (-E)	Изменить имя цепочки.	iptables -E OLD_CHAIN NEW_CHAIN
–flush (-F)	Очищает все правила текущей таблицы. Ко всем пакетам, относящимся к уже установленным соединениям, применяется терминальное действие ACCEPT — пропустить	iptables -F
–insert (-I)	Добавляет заданное правило в соответствии с номером.	iptables -I FORWARD 5 критерии -j действие
–list (-L)	Позволяет просматривать существующие правила (без явного указания таблицы - отображается таблица filter всех цепочек).	iptables -L

–policy (-P)	Позволяет устанавливать стандартную политику для заданной цепочки.	iptables -t mangle -P PREROUTING DROP
–replace (-R)	Заменяет заданное номером правило на заданное в критериях.	iptables -R POSROUTING 7 | критерии -j действие
–delete-chain (-X)	Удалить ВСЕ созданные вручную цепочки (оставить только стандартные INPUT, OUTPUT…)	iptables -X
–zero (-Z)	Обнуляет счетчики переданных данных в цепочке.	iptables -Z INPUT
–line-numbers	Указывать номера правил при выводе (может использоваться совместно с -L).	iptables -L –line-numbers
–help (-h)	Помощь	Iptables –help
-t таблица	Задает название таблицы, над которой необходимо совершить действие. В примере сбрасывается таблица nat во всех цепочках.	iptables -t nat -F
–verbose (-v)	Детальный вывод.	iptables -L -v
	Основные правила отбора пакетов	
–protocol(сокр. -p)	Определяет протокол транспортного уровня. Опции tcp, udp, icmp, all или любой другой протокол определенный в /etc/protocols	iptables -A INPUT -p tcp
–source(-s, –src)	IP адрес источника пакета. Может быть определен несколькими путями:Одиночный хост: host.domain.tld, или IP адрес: 10.10.10.3 Пул-адресов (подсеть): 10.10.10.3/24 или 10.10.10.3/255.255.255.0	iptables -A INPUT -s 10.10.10.3
–destination(-d)	IP адрес назначения пакета. Может быть	iptables -A INPUT –destination 192.168.1.0/24

	определен несколькими путями (см. –source).	
–in-interface (-i)	Определяет интерфейс, на который прибыл пакет. Полезно для NAT и машин с несколькими сетевыми интерфейсами. Применяется в цепочках INPUT,FORWARD и PREROUTING. Возможно использование знака +, тогда подразумевается использование всех интерфейсов, начинающихся на имя+ (например eth+ - все интерфейсы eth).	iptables -t nat -A PREROUTING –in-interface eth0
–out-interface(-o)	Определяет интерфейс, с которого уйдет пакет. Полезно для NAT и машин с несколькими сетевыми интерфейсами. Применяется в цепочках OUTPUT, FORWARD и POSTROUTING. Возможно использование знака +.	iptables -t nat -A POSTROUTING –in-interface eth1
	Неявные (необщие) параметры	
-p proto -h	Вывод справки по неявным параметрам протокола proto.	iptables -p icmp -h
–source-port(–sport)	Порт источник, возможно только для протоколов –protocol tcp, или –protocol udp	iptables -A INPUT –protocol tcp –source-port 25
–destination-port(–dport)	Порт назначения, возможно только для протоколов –protocol tcp, или –protemocol udp	iptables -A INPUT –protocol udp –destination-port 67
	Явные параметры	
-m state –state (устарел) он же -m conntrack –ctstate	Состояние соединения. Доступные опции: NEW (Все пакеты устанавливающие новое соединение) ESTABLISHED (Все пакеты, принадлежащие	iptables -A INPUT -m state –state NEW, ESTABLISHED iptables -A INPUT -m conntrack –ctstate NEW, ESTABLISHED

	установленному соединению) RELATED (Пакеты, не принадлежащие установленному соединению, но связанные с ним. Например - FTP в активном режиме использует разные соединения для передачи данных. Эти соединения связаны.) INVALID (Пакеты, которые не могут быть по тем или иным причинам идентифицированы).	
-m mac –mac-source	Задает MAC адрес сетевого узла, передавшего пакет. MAC адрес должен указываться в форме XX:XX:XX:XX:XX:XX.	-m mac –mac-source 00:00:00:00:00:0
	Дополнительные параметры	
	DNAT (Destination Network Address Translation)	
–to-destination	Указывает, какой IP адрес должен быть подставлен в качестве адреса места назначения. В примере во всех пакетах протокола tcp, пришедших на адрес 1.2.3.4, данный адрес будет заменен на 4.3.2.1.	iptables -t nat -A PREROUTING -p tcp -d 1.2.3.4 -j DNAT –to-destination 4.3.2.1
	LOG	
–log-level	Используется для задания уровня журналирования (log level). В примере установлен максимальный уровень логирования для всех tcp пакетов в таблице filter цепочки FORWARD.	iptables -A FORWARD -p tcp -j LOG –log-level debug
–log-prefix	Задает текст (префикс), которым будут предваряться все сообщения iptables. Префикс может содержать до 29 символов, включая и	iptables -A INPUT -p tcp -j LOG –log-prefix INRUT-filter

	пробелы. В примере отправляются в syslog все tcp пакеты в таблице filter цепочки INPUT с префиксом INRUT-filter.	
–log-ip-options	Позволяет заносить в системный журнал различные сведения из заголовка IP пакета.	iptables -A FORWARD -p tcp -j LOG –log-ipoptions

- SNAT — применить трансляцию источника в пакете. Используется только в цепочках POSTROUTING и OUTPUT таблицы nat.
- DNAT — применить трансляцию адреса назначения в пакете. Используется в цепочках PREROUTING и (очень редко) OUTPUT в таблице nat.

в скобках – сокращенный вариант записи

Основные цепочки межсетевого экрана Netfilter:

- PREROUTING — изначальная обработка входящих пакетов
- INPUT — для входящих пакетов, адресованных непосредственно локальному компьютеру
- FORWARD — для маршрутизируемых пакетов
- OUTPUT — для пакетов, исходящих с локального компьютера
- POSTROUTING — для окончательной обработки исходящих пакетов

Таблицы межсетевого экрана Netfilter:

- raw - используется для маркировки пакетов, которые не должны обрабатываться системой определения состояний. Содержится в цепочках PREROUTING и OUTPUT.
- mangle — содержит правила модификации IP-пакетов.
- nat - предназначена для подмены адреса отправителя или получателя. Данную таблицу проходят только первые пакеты из потока - трансляция адресов или маскировка (подмена адреса отправителя или получателя) применяются ко всем последующим пакетам в потоке автоматически. Поддерживает действия DNAT, SNAT, MASQUERADE, REDIRECT. Содержится в цепочках PREROUTING, OUTPUT, и POSTROUTING.
- filter — основная таблица, используется по умолчанию если название таблицы не указано. Используется для фильтрации пакетов. Содержится в цепочках INPUT, FORWARD, и OUTPUT.

Пример создания правила для межсетевого экрана

Рассмотрим две цепочки, задающие два основных правила Iptables — PREROUTING и FORWARD.

- iptables -t nat -A PREROUTING -i eth0 -j DNAT —to-destination 192.168.57.102
- iptables -A FORWARD -d 192.168.57.102 -j ACCEPT

Первая из них определяет первоначальную обработку всех пакетов, приходящих на адаптер eth0:

- -t определяет подключаемую таблицу, в данном случае — nat — для подмены адреса отправителя или получателя
- -A — выбор цепочки
- -i — входящий интерфейс
- -j — действие с пакетами, удовлетворяющими условию — в данном случае DNAT — подмена адреса получателя
- –to-destination — выбор адреса, на который перенаправляются пакеты
- Вторая определяет проброс пакетов через сервер:
- -A — выбор цепочки
- -d — выбор адресата
- -j — выбор действия

Web Application Firewall

WAF (Web Application Firewall) - это межсетевые экраны, работающие на прикладном уровне и осуществляющие фильтрацию трафика Web-приложений. Эти средства не требуют изменений в исходном коде Web-приложения и, как правило, защищают Web-сервисы гораздо лучше обычных межсетевых экранов и средств обнаружения вторжений.

Основные преимущества:

- Анализ поведения пользователя в используемом приложении;
- Позволяет осуществлять мониторинг HTTP трафика и проводить анализ событий в реальном режиме времени;
- Предотвращение вредоносных запросов;
- Распознавание большинства опасных угроз;
- Дополнение сетевых средств безопасности;
- Просматривать детальные отчеты об атаках и попытках взлома.

Задания к лабораторной работе

Часть 1

- Обновите привязки <sudo apt-get update>
- Установите web-сервер <sudo apt-get install apache2>
- Просмотрите список текущих правил iptables таблицы filter

```
sudo iptables -L
```

- Вы увидите, что список содержит три цепочки по умолчанию (INPUT, OUTPUT и FORWARD), в каждой из которых установлена политика по умолчанию (на данный момент это ACCEPT).
- С поможью команды <sudo iptables -S> данный список можно просмотреть в другом формате, который отражает команды, необходимые для активации правил и политик.
- Чтобы сбросить текущие правила (если таковые есть), наберите:

```
sudo iptables -F
```

- Цепочка INPUT отвечает за входящий траффик.
- Чтобы внести локальный интерфейс выполните:

```
sudo iptables -A INPUT -i lo -j ACCEPT
```

- Чтобы заблокировать весь исходящий трафик, кроме портов для SSH и веб-сервера, нужно сначала разрешить подключения к этим портам. В цепочку ACCEPT добавьте два порта (порт SSH 22 и порт http 80), что разрешит трафик на эти порты.

```
sudo iptables -A INPUT -p tcp -m tcp --dport 22 -j ACCEPT
sudo iptables -A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
```

- В данной работе мы не используем SSH. Так что удалим ненужное правило. Для этого:

```
sudo iptables -D INPUT -p tcp -m tcp --dport 22 -j ACCEPT
```

- Нужно добавить еще одно правило, которое позволит устанавливать исходящие соединения (т.е. использовать ping или запускать обновления программного обеспечения):

```
sudo iptables -I INPUT -m state --state ESTABLISHED,RELATED
-j ACCEPT
```

- Создав все эти правила, можно заблокировать все остальное и разрешить все исходящие соединения.

```
sudo iptables -P OUTPUT ACCEPT
sudo iptables -P INPUT DROP
```

- Просмотрите список правил

```
sudo iptables -L
```

- Добавим еще несколько правил для блокировки наиболее распространенных атак. Для начала нужно заблокировать нулевые пакеты <sudo iptables -A INPUT -p tcp --tcp-flags ALL NONE -j DROP>.
- Следующее правило отражает атаки syn-flood <sudo iptables -A INPUT -p tcp ! --syn -m state --state NEW -j DROP>. Теперь фаервол не будет принимать входящих пакетов с tcp-флагами. Нулевые пакеты, по сути, разведывательные. они используются, чтобы выяснить настройки сервера и определить его слабые места.
- Далее нужно защитить сервер от разведывательных пакетов XMAS <sudo iptables -A INPUT -p tcp --tcp-flags ALL ALL -j DROP>. Теперь сервер защищен от некоторых общих атак, которые ищут его уязвимости.
- С виртуальной машиины Kali linux проведите XMAS сканирование <sudo nmap -sX>, сделайте скриншот результатов.
- По умолчанию все не сохраненные правила действуют до следующей перезагрузки сервера; сразу же после перезагрузки не сохраненные правила будут потеряны. Самый простой способ загрузить пакет iptables-persistent <sudo apt-get install iptables-persistent>. Во время инсталляции пакет уточнит, нужно ли сохранить текущие правила для дальнейшей автоматической загрузки, если текущие правила были протестированы и соответствуют всем требованиям, их можно сохранить.

Часть 2

- Установите php <sudo apt-get install php libapache2-mod-php php-mysql>
- Обновите привязки <sudo apt update -y>
- Установите mod_security <sudo apt install libapache2-mod-security2>
- Перезапустите Apache2 <sudo systemctl restart apache2>

- Скопируйте и переименуйте файл modsecurity.conf

<sudo cp /etc/modsecurity/modsecurity.conf-recommended /etc/modsecurity/modsecurity.conf>

- Стандартный конфигурационный файл настроен на DetectionOnly, то есть, фаервол только отслеживает логи, при этом ничего не блокируя. Чтобы изменить это поведение, сначала перейдите в папку /etc/modsecurity

<cd /etc/modsecurity>

- затем отредактируйте файл modsecurity.conf: <sudo nano /etc/modsecurity/modsecurity.conf>

<sudo nano modsecurity.conf>

Если редактор nano не установлен, то установите его с помощью команды <sudo apt install nano>

- Найдите в файле строку: «SecRuleEngine DetectionOnly». И измените ее так: «SecRuleEngine On».
- Используйте CTRL+X для выхода, затем напишите y и нажмите Enter для сохранения.
- Выйдите из папки the /etc/modsecurity:

<cd>

- Перезапустите Apache

<sudo systemctl restart apache2>

- Загрузите последнюю версию OWASP ModSecurity Rules. Последняя версия Core Rule Set (CRS) для ModSecurity находится на GitHub.
- Установите Git, если он не установлен.

<sudo apt install git>

- Скопируйте CRS:

```
dejan@dejan-phoenixnap:~$ git clone https://github.com/SpiderLabs/owasp-modsecurity-crs
.git
Cloning into 'owasp-modsecurity-crs'...
remote: Enumerating objects: 10486, done.
remote: Total 10486 (delta 0), reused 0 (delta 0), pack-reused 10486
Receiving objects: 100% (10486/10486), 3.33 MiB | 1.39 MiB/s, done.
Resolving deltas: 100% (7687/7687), done.
dejan@dejan-phoenixnap:~$
```

<git clone https://github.com/SpiderLabs/owasp-modsecurity-crs.git>

- Откройте скачанную папку:

cd owasp-modsecurity-crs

- Скопируйте crs-setup file:

sudo mv crs-setup.conf.example /etc/modsecurity/crs-setup.conf

```
dejan@dejan-phoenixnap:~$ cd owasp-modsecurity-crs
dejan@dejan-phoenixnap:~/owasp-modsecurity-crs$ sudo mv crs-setup.conf.example /etc/mod
security/crs-setup.conf
dejan@dejan-phoenixnap:~/owasp-modsecurity-crs$
```

- Зайдите в директорию rules/:

sudo mv rules/ /etc/modsecurity

- Затем, проверьте security2.conf чтобы удостовериться, что правила ModSecurity загружены:

sudo nano /etc/apache2/mods-enabled/security2.conf

```
dejan@dejan-phoenixnap:~/owasp-modsecurity-crs$ sudo mv rules/ /etc/mod
dejan@dejan-phoenixnap:~/owasp-modsecurity-crs$ sudo nano /etc/apache2/
urity2.conf
```

- Проверьте, что там есть строки

IncludeOptional /etc/modsecurity/*.conf

Include /etc/modsecurity/rules/*.conf

- Закомментируйте строку IncludeOptional /usr/share/modsecurity-crs/*.load

#IncludeOptional /usr/share/modsecurity-crs/*.load

```
 GNU nano 4.8          /etc/apache2/mods-enabled/security2.conf
<IfModule security2_module>
        # Default Debian dir for modsecurity's persistent data
        SecDataDir /var/cache/modsecurity

        # Include all the *.conf files in /etc/modsecurity.
        # Keeping your local configuration in that directory
        # will allow for an easy upgrade of THIS file and
        # make your life easier
        IncludeOptional /etc/modsecurity/*.conf
        Include /etc/modsecurity/rules/*.conf

        # Include OWASP ModSecurity CRS rules if installed
        IncludeOptional /usr/share/modsecurity-crs/*.load
</IfModule>
```

- Перезапустите Apache

<sudo systemctl restart apache2>

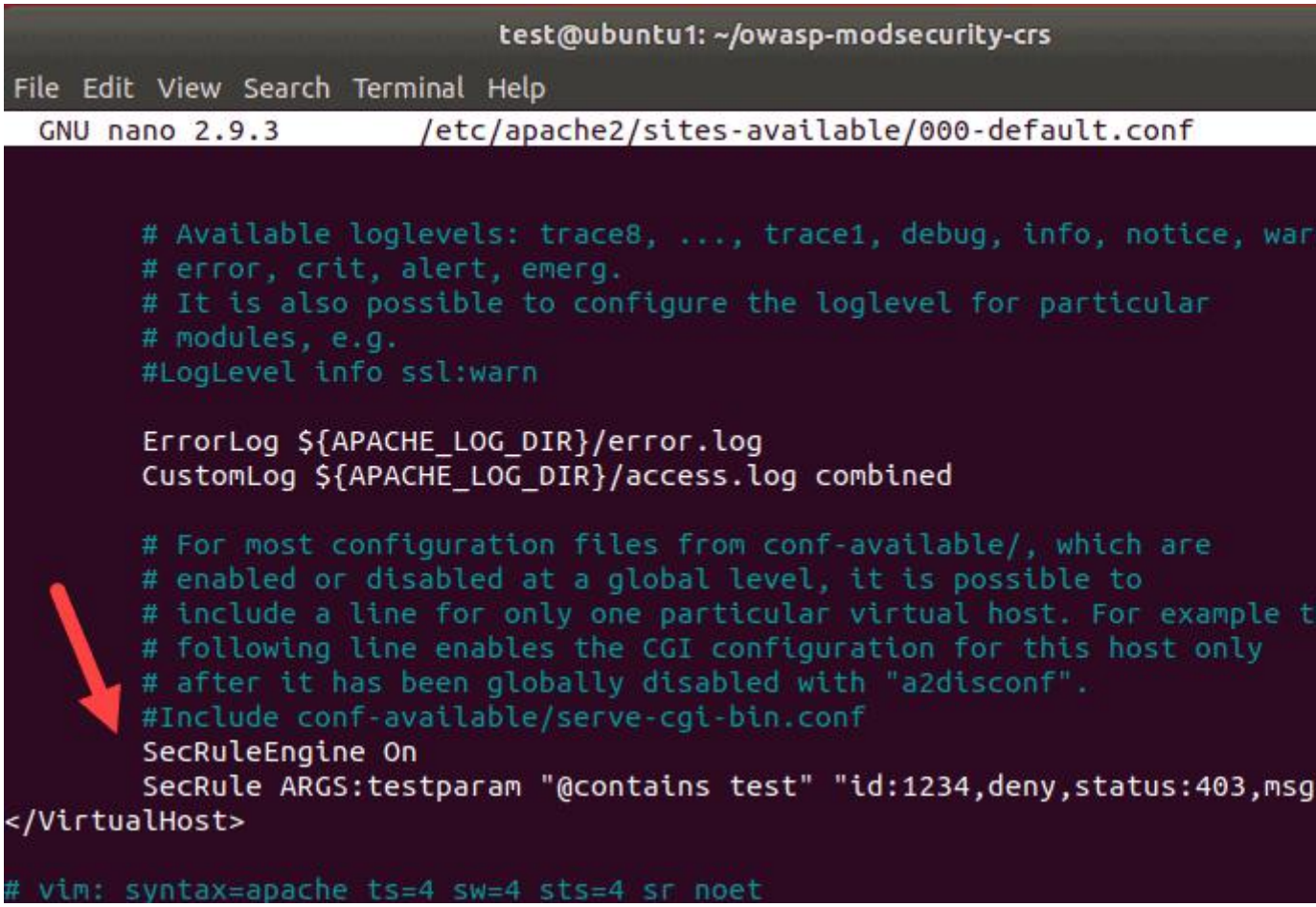
- Проверьте настройки Apache
- Откройте конфиг:

sudo nano /etc/apache2/sites-available/000-default.conf

- Найдите таг </VirtualHost> внизу страницы и добавьте перед ним строки:

SecRuleEngine On

SecRule ARGS:testparam "@contains test" "id:1234,deny,status:403,msg:'phoenixNAP test rule was triggered'"



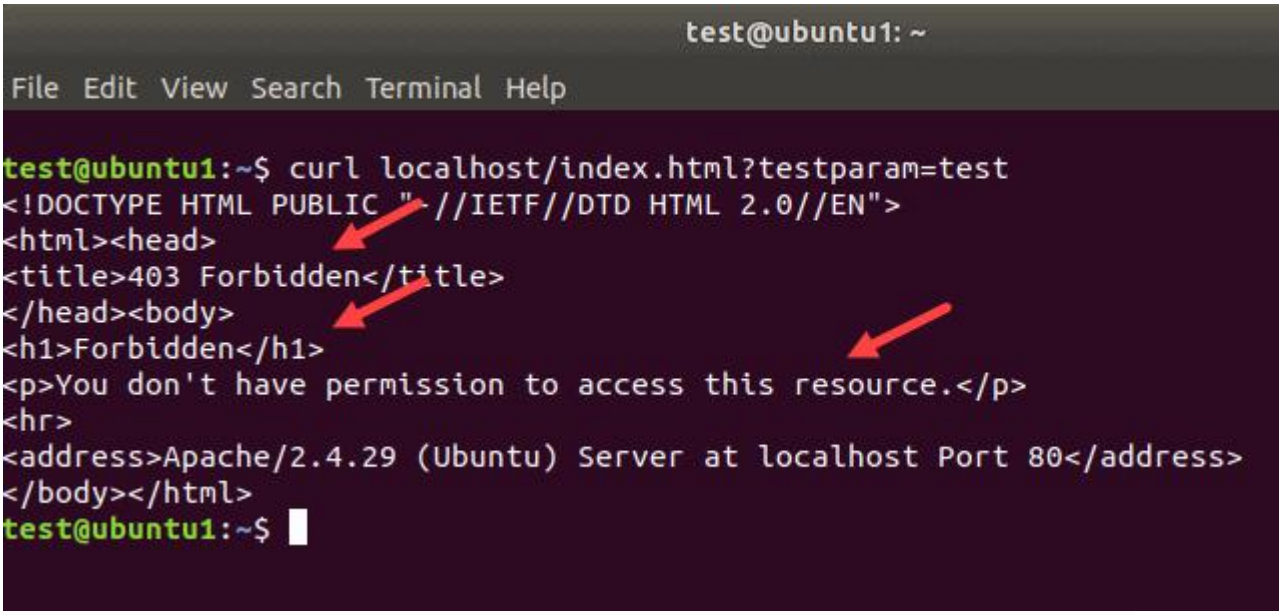
- Перезапустите Apache

<sudo systemctl restart apache2>

- Установите curl:

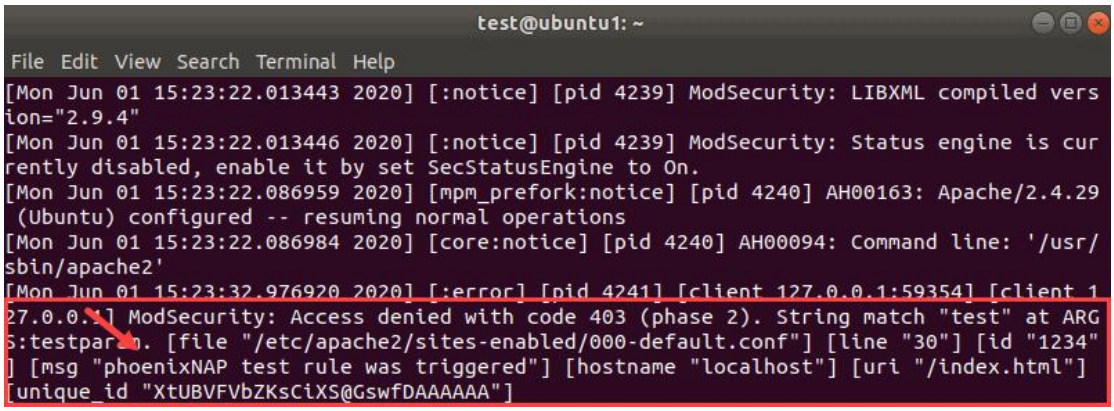
sudo apt install curl

- curl localhost/index.html?testparam=test
- Система ответит выводом страндартной веб страницы:



- Вы можете убедиться, что ModSecurity работает, если найдете code 403 в выводе ошибок Apache:

sudo tail -f /var/log/apache2/error.log

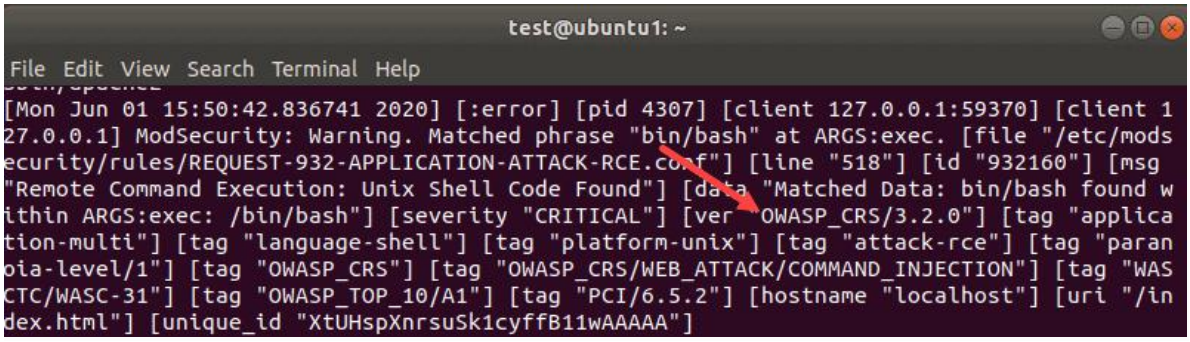


- Протестируйтет ModSecurity и OWASP CRS с помощью bash скрипта
- Введите следующую команду

curl localhost/index.html?exec=/bin/bash

- Вывод показывает ошибки
- Если посмотреть apache error.log, можно найти правило по которому был кик:

sudo tail -f /var/log/apache2/error.log



- Вывод показывает OWASP-related ModSecurity error message.
- Создае ModSecurity правила
- Внизу тестовый пример как можно использовать ModSecurity, чтобы блокировать нужные слова в PHP форме
- Создаем php файл в папке html с помощью команды

sudo nano /var/www//html/test.php

- Вводим туда следующий код:

```
<html>
<body>
<?php
if(isset($_POST['data']))
echo $_POST['data'];
else
{
?>
<form method="post" action="">
Enter text here:<textarea name="data"></textarea>
<input type="submit"/>
</form>
<?php
}
?>
</body>
</html>
```

- сохраните файл и выйдите
- Затем создайте свои правила для ModSecurity:

sudo nano /etc/modsecurity/modsecurity_custom_rules.conf

- Допишите следующие строки

SecRule REQUEST_FILENAME "test.php" "id:'400001',chain,deny,log,msg:'Spam detected'"

SecRule REQUEST_METHOD "POST" chain

SecRule REQUEST_BODY "@rx (?i:(enlarge|Nigerian|gold))"

- Если нужно, можно изменить контрольные слова.
- Сохраняем и выходим
- Перезапустите Apache

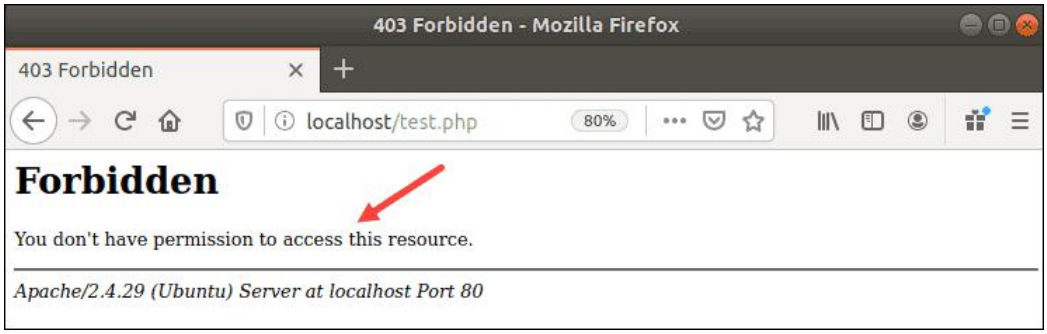
<sudo systemctl restart apache2>

- Устанавливаем firefox

sudo apt-get install firefox-esr

- Запускаем форму в web браузере
- localhost/test.php

- Вводим контрольные слова в форму: enlarge, Nigerian или gold.
- И получаем ошибку 403, ошибка доступа.



- Можете проверить логи /var/log/apache2/error.log file, чтобы проверить работу ModSecurity.

Вопросы к лабораторной работе

1. Что такое межсетевой экран?
2. Для чего используется межсетвой экран?
3. Принцип работы Netfilter.
4. Таблицы межсетевого экрана Netfilter. Для чего они используются?
5. Что такое правила межсетевого экрана?
6. Как создавать правила для межсетевого экрана утилитой Iptables?
7. Как сохранить правила для последующей автозагрузки?
8. Что такое Web Application Firewall?
9. Как настроить правила в WAF mod_security?

Составьте отчет о выполнении лабораторной работы.

Включите в него копии экрана и ответы на вопросы лабораторной работы.

Защита практической работы состоится на следующем занятии.